THE 3D CONSTRUCTION KIT

USER CLUB NEWSLETTER

ISSUE 1 JUNE 1991

Build your own Virtual Reality

CONTENTS

****

CORRESPONDENCE. All correspondence addressed to the User Group will be assumed to be for publication unless otherwise stated. If you wish for a personal reply to a letter then please enclose a S.A.E. otherwise replies will be published in the pages of the newsletter.

CONTRIBUTIONS are always welcome from members. Hints, tips, routines or anything else that you think would be of interest to other members are most welcome. It is your club so we want to hear from you. If there is anything in particular that you would like to see covered in the newsletter then please let me know.

****

HELPLINES

POSTAL HELP

If you wish to write in for help then please feel free to do so but please don't forget to enclose a stamped, self-addressed envelope and I will do my best to solve the problem.

TELEPHONE HELPLINE

If you need to telephone me for help then I will try to be available during the hours between 2pm and 7pm each weekday and will be delighted to hear from you and will, of course, do my very best to help with any problems you may have. As this telephone helpline is available only to members of the User Group, please do have your Registration Number handy and give your name when you call.

FACSIMILE

If anyone finds it more convenient to send a Fax then please feel free to do so. The telephone number is the same as that for the telephone helpline.

Please address all correspondence to:

Mandy Rodrigues
The 3D Construction Kit User Group

The 3D Construction Kit User Club is an independently run club but is under license to Incentive Software.

****

EDITORIAL

Welcome to issue 1 of the 3D Construction Kit User Group newsletter.

Future issues of the newsletter will contain a rather shorter editorial, about a page (if you can put up with my woffling for that long!), but as this first issue is rather special I thought it only fair to include something about myself and my "qualifications" for being editor/member-in-chief of the User Group, to introduce myself a little bit so you know who you are writing to or speaking to on the telephone. I have been the editor of Adventure Probe magazine for over three years and was a freelance writer prior to that. I have published three games of my own and several for other authors through my own software company and have been involved (devoted, obsessed!) with computers for quite a number of years now. I have an embarrassing number of computers of my own and have been privilaged to playtest all versions of the kit and, of course, wrote the manuals (with considerable help from the programmers, bless them!) so I feel confident that I can answer most of the problems that will come my way. If not, I know a man who can!

I am married with two children and am fortunate in that the whole family are computer crazy (hence the embarrassing number of computers in the family!). My husband deals with the photocopying side of things too and I don't know what I would do without him when it comes to producing all the newsletters. Well, enough about me, let us get on with more interesting matters now.

The aim of the User Group and the bi-monthly newsletters is to provide help, support and information on all aspects of the 3D Construction Kit. To exchange useful information and advice as we explore together the possibilities of one of the most exciting products to be released to date. I say "together" because, although I have been working with the 3D Construction Kit for quite some time now, there is still so much to be discovered and I will be learning alongside everyone else. As the 3D Construction Kit is so new there is so much to be explored and discovered and, I feel that the prospect is extremely exciting.

As this is the very first issue of the newsletter it is, obviously, written mostly by myself. I would like to remedy that if possible and ask that members do write in with their routines, suggestions, articles and the like so that as much information as possible will be available. So if you develop a routine or come up with an idea then do write in and tell me about it. You never know just how important that bit of information could be to another user. What may seem obvious to some may be very obscure to others (I speak from personal experience here as I do have a habit of going the longest route when the obvious shortcut is staring me in the face - the programmers at Incentive will know what I mean. They have been very patient with me and at times it has been rather like talking down an aircraft with all the pilots out of action and a passenger at the controls! - thanks boys!) And, of course, do feel free to write in for help too. I would love to hear from you, even if it is just to chat and tell me about yourself, so do please get in touch. I have already had quite a few letters, some asking for help and these will be dealt with in this issue.

The newsletter will develop and evolve over the coming months and I will try to ensure that each computer is covered in each issue. Once the 8 bit versions of the 3D Construction Kit are released there will

be at least one page devoted to each of the six versions with material especially relevant to each particular machine. In this way I hope to ensure that no-one feels left out and that there is something of interest for everyone in each issue.

Shortly I hope to launch our very own Public Domain library which, hopefully, will contain routines, borders, members games and all sorts of "goodies". It would also be nice if we could offer a playtesting service within the User Club so if anyone would like to offer their help in playtesting, evaluation etc on other member's games or environments could you let me know. Above all, please remember that this is YOUR newsletter and I would like you to let me know what you would like to see covered in the newsletter. Ideas, opinions (both good and bad) are welcome as it is only if I get a little feedback from you that I can make sure that you get exactly what you want from the User Club. Contributions will also decide the size of the newsletters. The minimum size will be as this issue but if there is a lot of material for a particular issue then it will mean a fatter issue with lots more information for everyone.

As this is the first issue there is very little in the way of letters for the letters pages so I would like as many of you as possible to write to me and let me know what you think about the Kit, you might even have a grumble or two, or some ideas that you would like to share with the other members. One of my very favourite sections of any newsletter or magazine are the letters pages so, if you feel the same, do take the time to put pen to paper (or pinkies to the keyboard) and get in touch. Your problems and ideas are also needed, remember that often a problem shared is a problem solved! It could well be that the area that is presenting difficulty for you is also causing someone else a problem so do let me know about it.

As I mentioned earlier, I have been working on the 16 bit versions of the Kit for quite some time now and I am still working hard, this time on the 8 bit versions and am becoming familiar with those versions at this very moment and I am happy to say that the 8 bit versions are just as exciting as the 16 bit versions. I am also busy, well, when I get the time, trying to work on my own environment and, like you, am thoroughly enjoying myself in the process. In fact, I get the sneaky impression that the 3D Construction Kit is taking over my life - I'm not complaining - who could!

I see the bottom of the second page approaching rapidly so I will leave you to get on with reading the rest of this newsletter now. I sincerely hope that you enjoy this first newsletter and that you find it useful and informative. If there is any particular aspect of the Kit that you would like me to cover in the next newsletter then please do let me know and I will do my best to please. One final word (does this woman never shut up!), I sincerely hope that you all get as much pleasure and enjoyment out of using the 3D Construction Kit as I do.

Right, that's all from me for this issue. I'll see you all again in the next newsletter. And meanwhile, please do get in touch with me for either help or to send in material for the newsletter.

Mandy

****

MORE ABOUT THE FREESCAPE SYSTEM

AREAS

The 3D Construction Kit uses an enhanced version of the FREESCAPE (tm) system. The system allows you to represent a virtual world that you can move around and interact with. This world is represented in three dimensions known as X, Y and Z. X is equivalent to left and right, Y is equivalent to up and down and Z is equivalent to near and far. The world is divided up into regions known as AREAS. Each area is like a box and has a set size. A coordinate of X-0, Y-0, Z=0 (View: 000.000.000) represents the nearest bottom, left hand corner of an area. The world is seen from a single viewpoint which occupies one unit, looking in different directions. The direction is represented by 3 Angles: X rotation represents looking up or down, Y rotation represents looking left or right and Z rotation represents looking sideways (tilting your head). These rotations are measured in degrees and 360 degrees means that you have turned full circle. ROT:000,000,000 means that you are looking straight ahead. Changing the Y rotation by 180 (ROT:000.180.000) by doing a U-turn, means that you are looking directly behind you. Changing the X rotation to 90 (ROT:090,000,000) means that you are looking straight down. An X rotation of 270 (ROT:270,000,000) means that you are looking straight up.

An AREA can be used to represent a room of a house or an "outdoor" region. Areas have no geographical relation to each other, but are tied together by ENTRANCES. An ENTRANCE has a position and View Direction. The user can be placed at an entrance position in a specific area by some form of trigger (ie. walking into a door. This could effectively "move" the viewer from a hall (on an area) into a room (another area)).

OBJECTS

Objects can be placed into an area to make the environment. These are solid and, as such, cannot be passed through when moving. Objects can have different sizes in X, Y and Z directions. Object position and size are measured in a different coordinate system to the 3D world.

There are several basic object types ("PRIMITIVES") which can be resized and combined to make larger, more complicated objects such as buildings, trees etc. The primitive objects consist of:

CUBOIDS: 3 dimensional boxes where the sides can be stretched and shrunk in 3 directions.

PYRAMIDS: These are similar to conventional pyramids but are truncated at the top (they are initially flat on top). A FREESCAPE pyramid, like the cube can be stretched and shrunk in 3 directions.

RECTANGLES: Flat (2 dimensional) boxes whose sides can be stretched and shrunk in 2 directions at any time.

LINES: Two points in 3D space joined together by a line. These two points may be moved in 3 directions.

TRIANGLES: Three points in 3D space forming a triangle. These points may be moved in 3 directions. Only triangular facets may be non

orthogonal.

QUADRILATERALS: Four points in 3D space forming a quadrilateral. This can be non rectangular.

PENTAGONS: As triangles but with 5 points.

HEXAGONS: As triangles but with 6 points.

SENSORS: A single point in 3D space which can be moved in 3 directions. Sensors have the ability to detect your presence within a defined distance and even fire at you!

Objects such as pyramids and triangles, which do not occupy a whole unit of space act like solid cubes when moving around the 3D world. For example it is not possible to stand on the slope of a pyramid. This cube is known as the object's BOUNDING CUBE.

Objects can exist in three states; VISIBLE, INVISIBLE or DESTROYED. Visible means that the object is present in the world as a solid form. Invisible means that the object is not present in the world but can be brought into the world by being made visible. Destroyed means that once an object has been destroyed it is invisible and cannot be made visible again until the world is reset.

Each individual side of an object can be coloured differently. The amount of colours available depend, of course, on which computer system you are using. But the first colour has a special property in that it is not drawn. This can be used to improve the speed of drawing the 3D world when used on object sides which are never seen, such as the underside of a house or the back of a door which is placed against a wall.

By painting all the sides of an object with this "Invisible" colour it is possible to have an object which cannot be seen but which is solid and cannot be moved through. Not that in this case the object's attribute will be set as VISIBLE even though it cannot be seen. This can be used to produce invisible barriers or invisible triggers on floors.

INTERACTION WITH OBJECTS

Events or reactions can be caused by interacting with objects within the environment. Objects may be triggerd to respond to being shot, activated (touched/pressed/manipulated). Collided with or walked on. However, an object will only be activated if the object is within a predefined distance known as the ACTIVATE RANGE.

SENSORS

A SENSOR is a special type of object that will detect your presence if you are within a set distance from it. This allows reactions to occur when approaching an object. A Sensor can also be set to respond by firing back at you at a given rate.

THE FREESCAPE COMMAND LANGUAGE

In order to interact with objects, check and actions have to be defined in the form of short programs called CONDITIONS. Conditions

are written using instructions which make up the FREESCAPE COMMAND LANGUAGE (FCL). FCL is a very simple but powerful set of commands which allow you to manipulate and respond to any occurances. There is also a bank of memory which can be used by the user to store and view information relating to the environment and events. Each memory cell is called a VARIABLE.

There are various categories of instructions in FCL:

VARIABLE MANIPULATION which is a set of instructions which act upon values stored in the variable memory cells. These include commands to add to and subtract values from variables, comparisons of values and setting a value.

OBJECT MANIPULATION which is a set of instructions which can alter the attributes of a specified object. There are instructions to visiblize, invisiblize, destroy and swap the visibility of objects.

CONDITIONAL INSTRUCTIONS are a set of instructions to allow the execution of segments of programs depending on the outcome of specific checks, such as, if a particular object is shot, if a particular object is collided with, if a Sensor has sensed you etc.

OTHER COMMANDS which are instructions to deal with functions such as printing messages, setting colours, playing sounds and setting the TIMER. The Timer is a device that can be set to trigger a set of conditions at a defined regular interval of time.

LIMITATIONS TO FREESCAPE

As FREESCAPE uses extremely complex mathematical algorithms to represent the 3D world on a 2D screen, there are a few basic limitations to what you can do:

Objects should not overlap. The BOUNDING CUBE of one object should not occupy the same 3D space as any other. This is especially true when you have GLOBAL OBJECTS or objects which are made visible which overlap with normal objects.

The number of visible objects in any one AREA is limited due to memory restrictions. Excess objects are not drawn. The maximum number of objects depends on the computer you are using.

As you can see, the limitations are very few considering just how powerful the program is

****

NEWS

Incentive have just announced that Domark will shortly be producing a compilation of games entitled 3D WORLD. The games that will form the compilation will be Driller, Total Eclipse, Castle Master and The Crypt. The compilation will be available on all six formats: Amiga, Atari ST, PC, Amstrad CPC, Commodore 64 and Spectrum. Details of price etc will be announced here when available. If any of you want to see just what is possible with the FREESCAPE system then these are a must. They are also extremely enjoyable games.

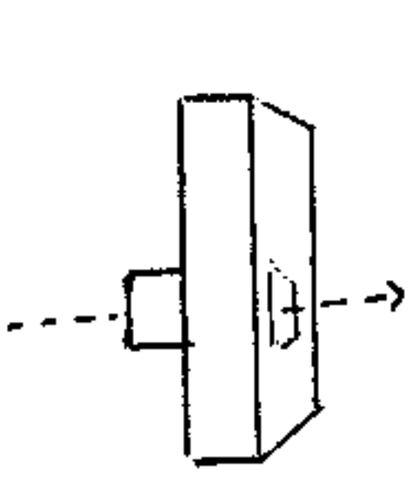
****

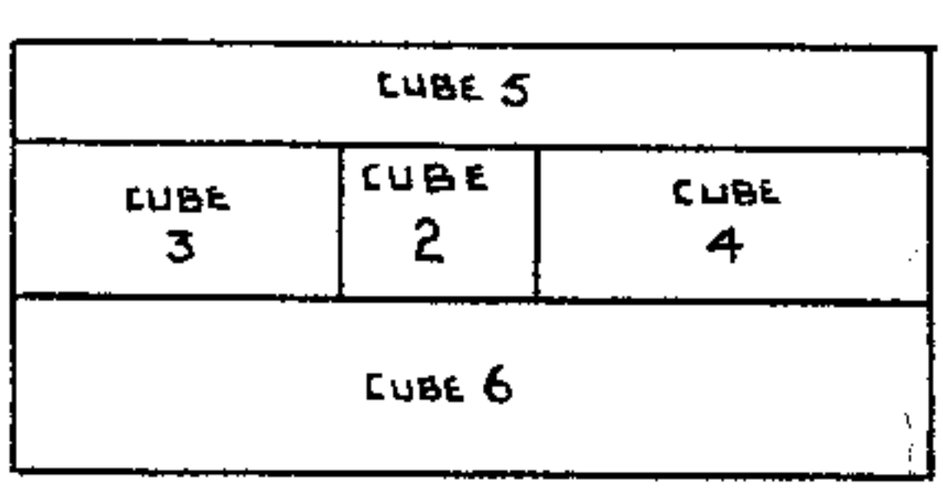
ANIMATION

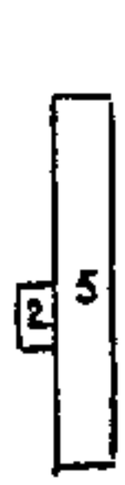
Some of the most powerful, and versatile, set of commands within the 3D Construction Kit are, of course, the animation commands. With these a wide variety of special effects can be created to really enhance your environments. This article will give some examples of how animation works and some simple examples to get you started.

The most important parts of the animation conditions are LOOP and MOVE. When we enter the LOOP command we are actually telling the system how many times we want the animation to be effected, ie. LOOP (10) tells the system to move the animation 10 times. The MOVE command tells the system the size and direction of the movement steps, ie. MOVE (40,0,0) tells the system to move towards the RIGHT in 40 size units. So putting the two commands together we are telling the system to move 40 units towards the right 10 times.

In the first example we are going to create the effect of a large block in a wall which can be pushed through to the other side. This could be useful if an object on one side of the wall were too high up to reach and we wanted the player to push a block through the wall from the ajoining room and so effectively providing a step on which to stand to reach said object. Firstly we would have to design the wall and block as follows:







Front view

Top view

The easy way to make the wall and block would be to firstly create a cube (cube 2) and copy this on both sides and above then stretch the top and side cubes (3, 4 and 5) to form a solid wall. Then stretch the "block" (cube 2) outwards to double its size to enable it to be pushed through the wall and appear on the other side. Go to the top Object Menu and select ATTRIBUTES and click on cube 2 and change from STATIC to MOVEABLE. Then enter the following CONDITION for Cube 2:

```
IF COLLIDED?
THEN STARTANIM (1)
ENDIF
```

Go to the top Area Menu and select CREATE ANIMATION. Go back to the top Area Menu and select EDIT ANIMATION. You will now see 001 ANIMATOR. Click on this and enter the following:

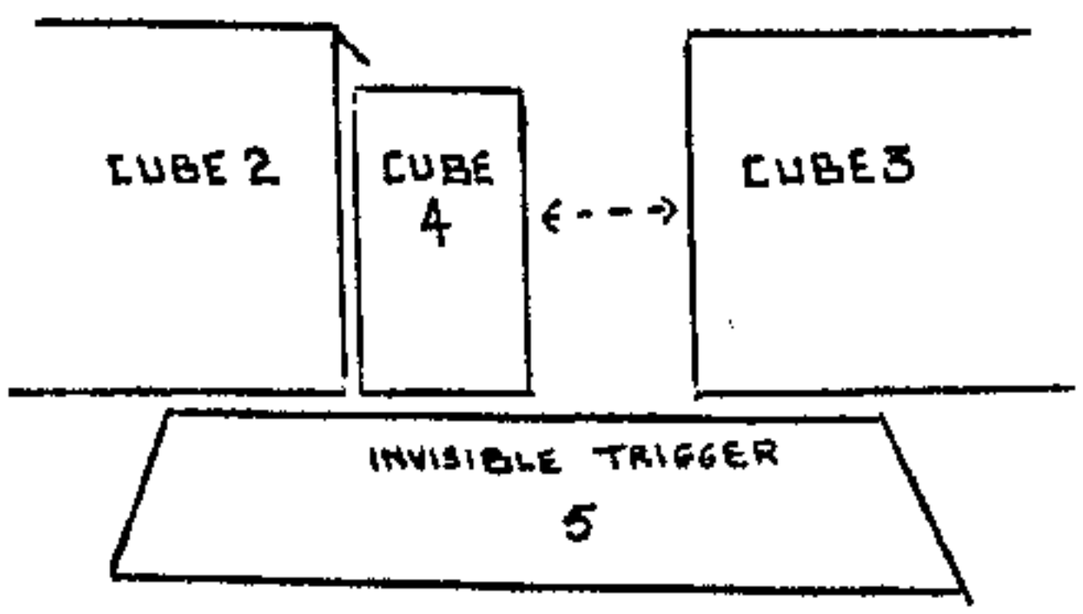
```
INCLUDE (2)
START
LOOP (15)
```

```
MOVE (20,0,0)
AGAIN
```

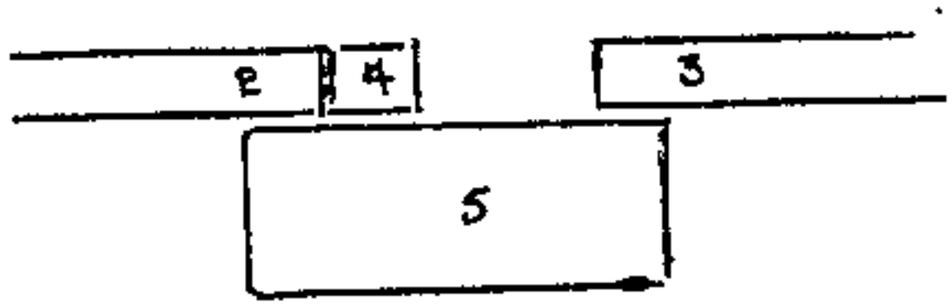
Now if you walk into the cube, effectively pushing it, you will see it glide into the wall and if you walk around the wall to the other side you will see that it has now been pushed through to the other side. Of course the same effect could be done using the "Activate" command.

(*Thanks to Bob Rodrigues for this example*)

In the next example we will create a gap in a wall with a moveable block to one side but have an invisible trigger on the floor in front of it so that as the player approaches the gap (doorway?) the block will start to move backwards and forward between the opening. To make life a little more difficult for the player we would have another condition on the moving block which meant that if they collided with it then the game would end and reset. The only way through the gap would be to get the timing of their movements exactly right. Firstly we would have to design our section as follows:



Front view



Top view

Make two walls with a gap between (cubes 2 and 3) and put the moveable block (cube 4) between them and up against cube 2. Don't forget to make this moveable via the Objects Attributes as in the previous example. Now create a rectangle (5) and using the EDIT option turn it and bring it down so that it is lying flat, rather like a carpet, in front of the gap. Colour this rectangle with the INVISIBLE colour which makes it invisible to the player even though it is really there. Enter the following condition for Rectangle:

```
IF COLLIDED?
THEN STARTANIM (1)
ENDIF
```

Create an animator, select it for editing and enter the following conditions:

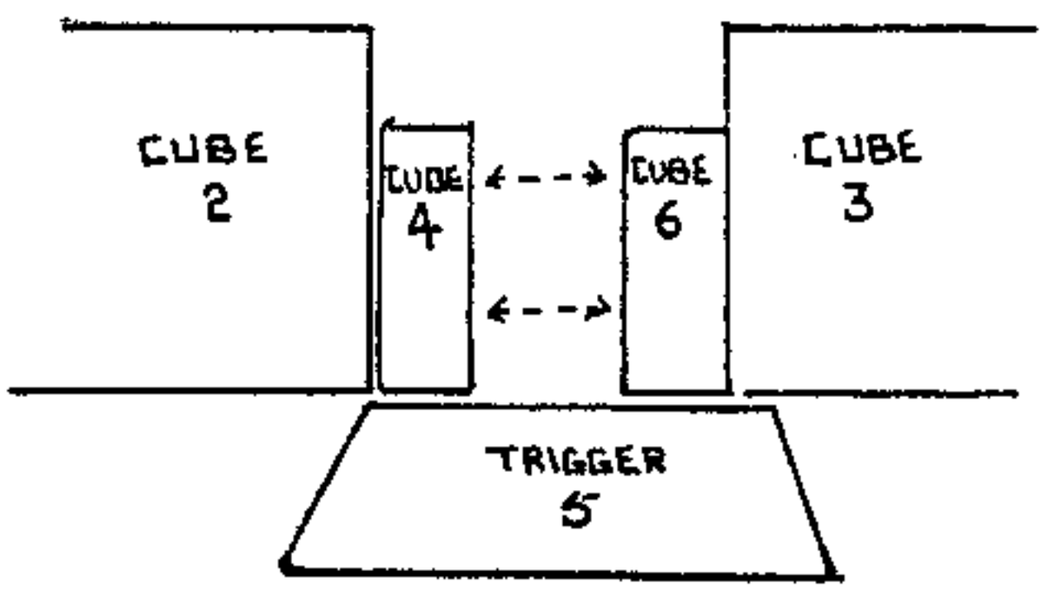
```
INCLUDE (4)
START
LOOP (15)
MOVE (30,0,0)
AGAIN
LOOP (15)
MOVE (-30,0,0)
AGAIN
RESTART
```

Now creat a condition for cube 4 as follows:

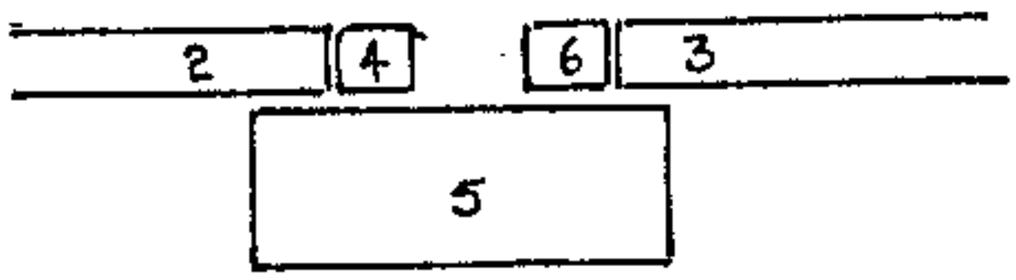
```
IF COLLIDED?
THEN ENDGAME
ENDIF
```

Now try this out. Walk towards the gap in the wall and when you step on the "trigger" (rectangle 5) the animation should start. Try and walk past the moving block without touching it then try once more but touching it this time and see how the game immediately resets to the original state. Of course, when creating the finished game we would design an END SEQUENCE which would give a more polished effect than a reset.

Of course we could give the player more of a headache if we were to have two cubes moving backwards and forwards between the walls, meeting in the middle! For this we would need to create two cubes instead of one within the gap and make sure they were both moveable. This should look something like this:



Front view



Top view

We would then edit the condition for our "trigger", rectangle 5, as follows:

```
IF COLLIDED?
THEN STARTANIM (1)
STARTANIM (2)
ENDIF
```

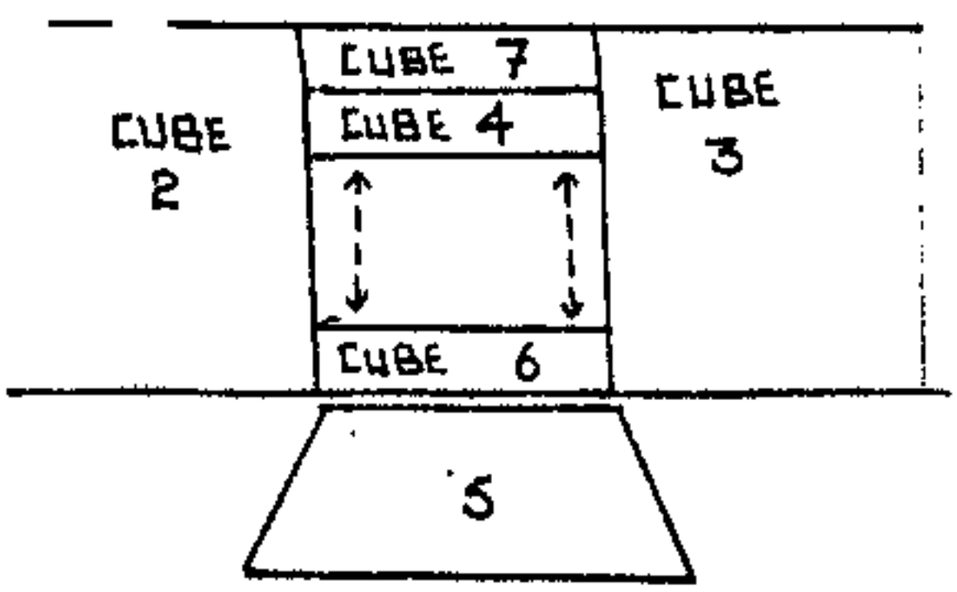
We would now need two animators, one for Cube 4 and the other for Cube 6. Our first animator can be left the same and we would enter the following condition for Animator 2:

```
INCLUDE (6)
START
LOOP (15)
MOVE (-30,0,0)
AGAIN
LOOP (15)
MOVE (30,0,0)
AGAIN
RESTART
```

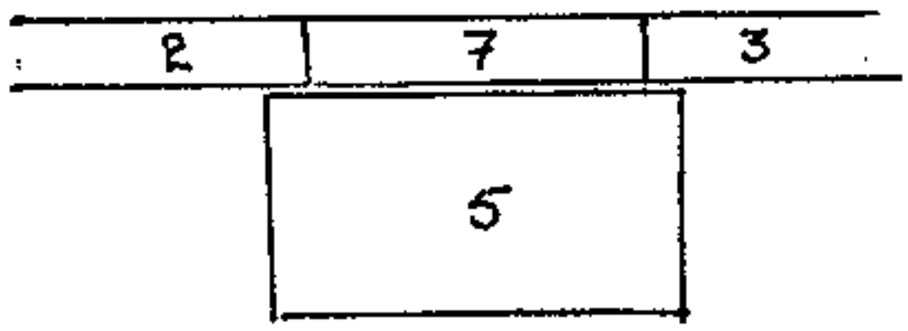
Note the position of the first "minus" symbol. It is opposite to the first Animator so that they move in opposite directions!

Now test this out to see the effect. If necessary the gap can be widened via the EDIT icons to accommodate the two cubes. Experiment a little with the loop command and the first number after the move command to see the effect.

In the next example we will do something slightly different. We will have a hole in the wall with two cubes, one at the top and one at the bottom, which when animated will move up and down within the gap. Design your setting like this:



Front view



Top view

Create two Animators and enter the following condition for Animator 1:

```
INCLUDE (6)
START
LOOP (15)
MOVE (0,-10,0)
AGAIN
LOOP (15)
MOVE (0,10,0)
AGAIN
RESTART
```

Enter the following condition for Animator 2:

```
INCLUDE (4)
START
LOOP (15)
MOVE (0,10,0)
AGAIN
LOOP (15)
MOVE (0,-10,0)
AGAIN
RESTART
```

Note once again the position of the "minus" symbol and also that we are now using the Y coordinate to move upwards and downwards instead of from side to side.

Walk towards the gap and see how this animation works. Of course, if you wished you could INCLUDE more than one object if, for example you wanted to create a portcullis to move upwards and downwards within a castle entrance. Many different kinds of "traps" and "problems" can

be created for the player using this kind of animation.

Animation can be used to enhance the environment as well as for obstacles to the players progress. Lots of pleasing effects can be obtained. For example a swing in a park that really swings. For this example we will not design a full park but will show simply how the effect can be created.

Use the CLEAR ALL option under the File Menu and create a cube. Make this moveable via the Objects Attributes menu. We will use the floor as a trigger this time for simplicity so enter the following condition for Cube 1:

```
IF SHOT?
THEN STARTANIM (1)
ENDIF
```

Create an Animator and enter the following condition:

```
INCLUDE (2)
START
LOOP (15)
MOVE (10,10,10)
AGAIN
LOOP (15)
MOVE (-10,-10,-10)
AGAIN
RESTART
```

Now shoot the floor and see the effect.

Of course in the final "park" scene it wouldn't be a cube that was being animated but the seat of the swing. But is is a good example to show the smooth movement of the animation in three directions. Experiment with the Loop and the Move commands to see the different effects that can be obtained.

These are just a few examples of how animation works and are designed purely and simply to get you started. Many other pleasing effects can be obtained using animation. A flickering torch in a sconce on a dungeon wall in a castle. A bird flying across the sky. A fish swimming in the sea (as in the Kitgame with the shark and the little golden fish beneath). Water dripping from a tap in a bathroom. This would be particularly effective if we were to include sound effects as follows:

```
INCLUDE (2)
START
LOOP (15)
MOVE (0,20,0)
SOUND (4)          - (Amiga PING sound effect)
AGAIN
RESTART
```

A portcullis coming down with full clanking sound effects can be quite atmospheric, especially if it is coming down on your head! Have fun and do experiment with animation and please remember to send in your routines for inclusion within the newsletter to help other members - and me too!

****

BORDERS

One of the readers told me that the information on making borders for your final game is rather sparse in the manual. Of course, it wasn't possible to cover everything in depth within the manual, (which is why the User Club comes in very handy), and I propose to remedy this now with a few hints and advice on creating them.

The very first thing you should do, before you start creating your masterpiece, is to take a good look at your game and to make some notes. I look at all aspects of my game and note down such points as, will there be a scoring system, will there be treasures to be found and all sorts of things like that? Games can include messages to the user which come up at various points during play and, most importantly, which forms of movement the player will be able to use.

Make a note of all these items as space and layout is very important when planning your border. Icons will have to be designed to take care of all movements you think necessary and spaces for messages, score tables etc have to be allowed for. You will notice in the border provided for the Kitgame that some of the movements which are accessible within the main Edit screen are not catered for so do ensure that you have everything well planned before you even load in your Art Package.

One of the very first things I did when I wanted to create my first border was to load in my Art Package (in this case it was the Deluxe Paint III for the Amiga), and actually load in the border from the Construction Kit into Deluxe Paint III to see exactly how it had been created. I then (rather cheekily) swiped some of the icons and saved them as bushes for later use, just to get the feel of how things worked. Later, of course, I was able to create my own borders just the way I wanted them. For those, like myself, who are not especially artistic, it is very considerate of Incentive to provide a couple of ready made borders so that we can get to work straight away.

It is extremely important, and time saving in the long run, to read the manual very carefully to make sure that you know exactly what format is acceptable to your machine. I know from my own experience how frustrating it is to work for hours on a border and then not be able to load it into the system because I didn't check the number of pixels, number of colours and format and had to start work all over again! The format for the Atari ST and Amiga are 320 x 200 pixels, 16 colours, Lo-Res and NTSC format. On PC CGA it is 320 x 200 pixels with 4 colours and on PC EGA it is 320 x 200 pixels in 16 colours. On the PC the borders must be in IFF format only and have an extension of .LBM. If you are unable to make an NTSC with your art package then cut the screen as 320 X 200 pixels.

Although it may seem more sensible to position your border with the main control section directly beneath the View Window, you can, of course, have this anywhere on the screen you wish as the ability to Set the View Window is very flexible. Consider what type of game you are creating and design a border accordingly. Perhaps you are thinking of designing an adventure in which the player has to explore ancient Egypt and there are plenty of desert scenes. In this case it would be worth while, perhaps, to incorporate Hyroglyphics into your border, or scenes representing the ancient Egyption Book of the Dead. If, perhaps, you were creating a game set in medieval times some shields

and crossed lances etc. could be incorporated into the border. All this adds to the overall atmosphere and presentation of the finished product.

I mentioned an Egyption scenario in particular because I was talking to an artist friend of mine who provided the excellent graphics for Level 9's later releases and for other software companies, and he remarked that the 3D Construction Kit would be the ideal medium for creating such scenarios with deserts and pyramids etc. But of course, with a program such as this there is so much scope for all types of scenarios. If an environment were set in outer space, for example, the border created could be as fantastic as you wish. In fact, anything goes in this case, providing that all movements and messages are catered for and that the player isn't dazzled by green, purple, pink and yellow controls!

If the environment were to represent the view from some form of vehicle then the obvious border would be one to represent the inside of that vehicle with the controls on display and the environment made to fit into the windscreen or porthole, or whatever, of the chosen vehicle.

On the Atari ST versions of the Kit a split screen has been used and it is rather important that we remember this when creating our borders for that machine. If lots of different colours are used in creating our borders on this machine the overall effect can be quite spoiled by the split screen, as the top half of the border can be lost altogether or the colours changed unless care is taken when designing the border. One easy way of overcoming the problem would be to ensure that the border consists of a control panel only or that the sides of the border only reach half way up the screen. But if a full border is required the careful selection of colours can overcome the problem. Make sure that the colours chosen are compatible with the colours used within your environment, in particular, the "sky" colour. In this way the border which took a lot of effort will remain exactly as intended. The split screen is peculiar to the Atari ST only and owners of other versions of the 3D Construction Kit do not have this problem and can use a full screen border.

Another point worth mentioning here is that care should be taken to ensure that the border itself doesn't "overshadow" the environment. It would be a shame if your environment were spoiled by an overlarge and overpowering border with overbright colours. Try to picture your environment and the shades used and to select colours which compliment and enhance your finished product rather than clash horribly and spoil the overall effect.

For those of you who are particularly artistic perhaps it would be a nice idea if you were to donate some of your borders in the form of Public Domain software, which can be distributed through the club. This, I am sure, would be of particular interest to those of us who get all "fingers and thumbs" when it comes to using art packages. So if you do have a border that you are particularly proud of and would like to share with other members, do let me know.

Finally, if anyone has any ideas on creating borders, or any other aspect of the 3D Construction Kit for that matter, do please feel free to write them down and send them in to me for publication within the newsletter.

****

VARIABLES

To many people who are not too familiar with programming languages, the use of variables can seem very complicated. Indeed, at first they do seem rather daunting but I will try to give a few examples here of how they work to enable you to feel confident about them. Remember that they are very useful tools and with a little understanding of how they work you can really create interesting and involved environments and games.

In this first example, which is a very simple way of handling a variable, we will first create two cubes, cube 2 and cube 3. Via the object attributes menu we will make cube 3 invisible both in the initial and current status. This can easily be done by creating a cube and copying it to the left and using the edit icons seperate them slightly.

Now we will enter a condition for cube 2 as follows:

```
IF SHOT?
THEN ADDVAR (25,V40)
IF VAR=? (100,V40)
THEN VIS (3)
ENDIF
ENDIF
```

Note the two ENDIF's at the end of the command. This is because we have used two IF commands. What we are doing is to add the value 25 to variable 40 if the cube is shot and then saying if variable 40 equals 100 then make cube 3 visible.

Now shoot cube 2 four times (25 times 4 = 100) and cube 3 should become visible within the view window.

SYSTEM VARIABLES

We can make use of the variables which are reserved for use by the FREESCAPE system itself which can make life much easier for us. Take the above example for instance. We could do exactly the same thing and create the exact effect by using variable number 21 which actually counts the number of shots fired for us. In this case we would set up the two cubes in exactly the same way as before, making cube 3 invisible but instead of entering the condition for cube 2 (make sure you delete those conditions before going any further) we would create an Area Condition and enter the following:

```
IF VAR=? (10,V21)
THEN VIS (3)
ENDIF
```

Now shoot cube 2 ten times and see how cube 3 becomes visible. Of course, using this method we are just checking how many times we have shot but not WHAT we have shot so if you wish a specific object to be shot to achieve an effect it would be better to use the former object condition.

Making use of the system variables can be useful in a number of ways. We could use variable 12 to check the times the player has been crushed and then kill him off. This could be done very simply and quickly for our example. Clear all via the File Menu and create a

cube. We will use an animation for this so create an animator and edit it as follows:

```
INCLUDE (2)
START
LOOP (15)
MOVE (0,20,0)
AGAIN
LOOP (15)
MOVE (0,-20,0)
AGAIN
RESTART
```

Make cube 2 moveable via the attributes and select the "ground" cube (1) as the trigger by entering the following condition for cube 1:

```
IF SHOT?
THEN STARTANIM (1)
ENDIF
```

Now create an Area Condition and enter the following:

```
IF VAR=? (10,V12)
THEN ENDGAME
ENDIF
```

Shoot the ground and when the cube starts to glide up and down, make your way until you are standing directly underneath it. Now wait until you have been "crushed" about 10 times (remember that each time the cube comes up and down it counts as crushing you twice!) and the game should reset.

Of course these examples are very simple but with a little imagination splendid effects can be obtained using the right environment and objects.

VARIABLES AND SENSORS

In the next example we will use a SENSOR and allow it to shoot us until we are killed (well, with a little imagination you know what I mean). Create a Sensor and make it invisible via the attributes. Use the ground as a trigger to make it visible when the ground is shot - see the previous examples for how to do this. Then create an Area Condition and enter the following:

```
IF VAR=? (10,V11)
THEN ENDGAME
ENDIF
```

We use variable 11 because it counts the times that YOU have been shot. See page 61 of the manual for the list of variables and their uses.

Select RESET before we try this as when we created the sensor it shot at us before we could make it invisible. Now shoot the ground and count the number of times that the sensor shoots us. After 10 times the game should reset.

Before ending this article with another couple of examples, it would be helpul if you read the paragraphs which you will find just below

the variable list on page 61 as we will use variable 20. What we will try and do now is to create a situation in which the player cannot shoot until they discover and take the gun! For our purposes we will use a cube yet again to represent the gun (I said you would need a bit of imagination for this didn't I?). As the first General Condition is only called after a reset we will create it and enter the following:

```
SETVAR (V20,9)
```

Create the cube (gun) which will be cube 2 and enter the following Object Condition for cube 2:

```
IF ACTIVATED?
THEN INVIS (2)
ADDVAR (6,V20)
ENDIF
```

Note that before you "get" the gun (by using the right mouse button when close to it), you cannot shoot at all but once you have the gun you can!

ANIMATING THE PLAYER

Routine by Bob Rodrigues.

By using the first set of variables used by the system we can achieve some spectacular effects as can be seen in the Kitgame when the player enters the boat and is transported over the sea to the lighthouse. For our boat we will use yet another cube which we will flatten to make something resembling a skateboard without wheels but it is, for our purposes, a boat. Raise the "boat" just a little way off the ground. Create a river by creating a blue rectangle and stretching it across the ground a little distance away from you. Don't forget to make the "boat" moveable via the attributes! Enter the following condition for the "boat":

```
IF COLLIDED?
THEN STARTANIM (1)
ENDIF
```

Create an Animator and enter the following:

```
INCLUDE (o) (the number of our boat)
START
LOOP (100)
MOVE (0,0,20)
ADDVAR (20,V2)
AGAIN
```

If you move forwards until you are standing on our "boat" you should, if all goes well, find yourself sailing smoothly across the river on the boat! We used Variable 2 because this controls the Z position within the View window. Experiment a little by using the variables 0 and 1 to see the different effects that can be achieved.

I hope that this will help those of you who were unsure about the use of variables to get used to how they work and how you can use them to your advantage. I would be most interested to hear about any special routines or effects that you achieve for future issues of the newsletter.

****

LETTERS

Dear Mandy,
I have just received my receipt after joining the 3DKC User Club. First I would like to say that your club is a very good idea, I know it will help me a lot on my Amiga version. I do need some help though. How do I make use of the Fly 1 and Fly 2. When tried in editing mode they make no difference. In the manual it says you can set Cameras. How? In early reviews (before 3DCK was released) we were told that you create your own environment and "drive" around in it. How could you enter a vehicle and drive around? The only way I can see is if you could load in borders during the game, is this possible? I have some ideas that may be useful, why don't you ask people to send in finished games on disk and start a mini PD club (Public Domain), one for each format and do a small review on each in the newsletter. You could charge, say, £1.99 per disk including P and P.

ROBERT DAVIS, Caterham, Surrey.
(Amiga version)

(Try using Fly1 and Fly2 when in Test to see the results, Robert. The Cameras are not available for use in Test or within stand-alone environments but can be selected using the Mode Icon within the editor (left mouse button to select and right mouse button to set). I'm afraid it isn't possible to load in borders during the game in a stand-alone environment but if your border were designed to represent the inside of a vehicle it would give the appearence of moving around the environment. You can actually move the player around the environment if they enter some form of "open" vehicle though. Check out the article on Variables for information on how this can be achieved. I too have been thinking about a PD library and would be more than happy to organise this. Its success would depend upon the response from the members though. Anyone else care to let me know their feelings on the matter?.....Mandy)

==========

Dear Mandy,
I was thoroughly delighted with the 3D Construction Kit when it arrived. Full marks to Incentive for producing such a fantastic program. I have always been fascinated with the idea of computerised 3 dimensional images and this is by far and away the most user friendly program of its kind that I have seen. I am enclosing my registration slip and my application for membership of the User Club which, I hope will provide me with a little extra help and support whenever I need it.

DOUGLAS BROAD, Luton, Beds
(Atari ST Version)

==========

Dear Mandy,
I am enjoying working with the Construction Kit and also playing the Kitgame (purely to get to know how the system works of course, ha ha). I don't seem to be making much progress though, especially in getting the gold bar that sits in front of the monster er... dragon?... dog? that I meet when I open the door just opposite the door to the shop. I managed to shoot it and it stopped shooting back at me but that's as far as I got, I still couldn't get the gold bar. What have I done wrong do you think?

SIMON CROSS, Stafford.
(Amiga Version)

(The monster will stop firing at you if you manage to hit the spot where the sensor lies, around the nose area. The way in, though, to the gold bar lies in another direction. Go into the Kiosk and go behind the counter to the left of the screen until you see an opening. Lower yourself until you can enter the passage you see there and shoot the trapdoor. Walk towards it until you fall down and make for the steps you will see. These lead up into the mouth of the monster. Jump down and you can get the gold bar. You can also turn the "force field" off via a panel on the wall that you will find on the right side of the screen. I hope this will help you to make a bit more progress. If you need any more help on this then please do let me know.....Mandy)

==========

Dear Mandy,
I have purchased your 3D Construction Kit from the Home Computer Club and am registering my copy with you. Is it possible, however, that I can upgrade my program from version 1.00 to a higher one even if I have to pay a little extra. Please let me know.

G.SEWELL, Spratton, Northants.
(PC Version)

(At present there is only version 1.00 available but, version 1.2 is being finalised at the moment, although this contains no new features. More details about this and other versions will follow as I receive them as well as any other news about the program, utilities, upgrades etc..... Mandy)

==========

Dear Mandy,
I am having difficulty in installing the 3D Construction Kit onto my Hard Disk. I have entered A:INSTALL but it just doesn't seem to work. Can you please help me out?

DAVID BUCKNALL, St Helens, Merseyside
(Atari Version)

(You seem to have the command wrong, David. The correct way of entering the command you want is A: (or B:) (RETURN) then INSTALL (RETURN). I think that should do the trick.....Mandy)

==========

Dear Mandy,
When trying to make a stand-alone game using the MAKE command, how do you load in the border when you want to run the game? I have created (File),RUN but when you RUNEGA (File).RUN the border obviously isn't loaded.

MARTIN REDWAY, Sidcup, Kent
(PC Version)

(To make a stand-alone game which contains the border of your choice you must first load the border into the Kit together with the Datafile containing your environment. When you then use the MAKE the border data will be saved along with the datafile and the sound samples. If you get any further trouble then please let me know.....Mandy)

****

More letters next issue - Mandy

PROBLEM PAGE

I have an Amiga 500 computer and am enjoying working with the 3D Kit very much indeed. I do have a problem though. When I am saving or loading my datafiles from disk I sometimes get a DOS error and am unable to access the drive. Can you help me sort out what the trouble is?

ANGELA BISHOP, Stoke-on-Trent, Staffs.

(Well, Angela, as you do not say which DOS error comes up it is difficult to say exactly what is going wrong. I suggest you check out page 65 of the manual where you will find a list of the more common errors listed. It is worth noting, though, that many DOS errors can be corrected by clicking on the top DF0: in the File Selector. It usually works for me.....Mandy)

==========

I noticed that when I had completed one area and filled it with a nice design of a house complete with windows, guttering, garden with plants and trees etc, that the movement became quite slow. Can you tell me why this happens. Is it usual or have I discovered a bug?

BRIAN WILLIAMS, Dyfed, South Wales.

(It looks as if you have rather too many objects in that particular area, Brian. When there are a lot of them it does slow movement down quite a bit. If you ever saw CASTLE MASTER by Incentive you would be surprised to learn that they only had around 40 object in each area to ensure speedy movement throughout the game. There are quite a number of things that can be done to overcome the problem, though. One is to make all surfaces of objects which are not visible in the View window (i.e. bases of cubes), invisible. A clever use of objects is also quite useful. Suppose you had a window on your house that was made up of a blue rectangle for the pane and four edited cubes for the surround it would be much more economical to use two rectangles one larger than the other for the frame and a smaller rectangle on top of this for the pane. In this way you would use only two objects instead of five. If anyone else has any useful tips for saving memory in this way then please do get send them in..... Mandy)

==========

I am thoroughly enjoying the 3D Kit, it is fantastic! Only one irritation with the system - why is it that sometimes when trying to position a rectangle for example, against another object such as a cube, it vanishes into or behind the cube and has to be brought back into view once more?

JOHN SCHOFIELD, St Helens, Manchester.

(The way round that problem is to change to FINE step. Using FINE usually does the trick as the number of pixels used in movement is smaller and you should be able to position rectangles against other objects without any difficulty. While on the subject, it is useful to note that using FINE step will also shrink objects a lot further than the USER step.....Mandy)

Any other problems? Send them in and I'll do my best to help.

****

HINTS AND TIPS

NAMING OBJECTS

When designing your environments and a lot of objects are being created it is much easier to find the object you want to edit, or include within a group, if you can see exactly what the object is at a glance. It is much easier to discover which is the "doorway" rather than rectangle (32), especially if the object you want is obscured by other objects within the View window. This is simply done by selecting ATTRIBUTES and selecting the object you want to name. Click on the NAME of the object and an inverse cursor will appear. Delete whatever is written there, except the number and enter DOORWAY. Now when creating your environment you should find life a lot easier.

==========

TIME SAVING

If there were a large amount of objects within one area such as the upper and lower floors of a house, movement could become quite slow. In this case it would be a good idea to make all the object on the upper floor of the house invisible whilst exploring the lower floor. Then make the lower floor invisible whilst exploring the upper floor. We could use the staircase as a trigger to toggle the INVIS and VIS flag on all the objects. This should speed things up considerably. To do this we would start all the objects on the upper floor as INVISIBLE and use some sort of trigger, such as when one of the steps on the staircase is collided with, to make the upper floor objects VISIBLE and the lower floor objects INVISIBLE, and of course another trigger could be used to reverse the process when descending the staircase once more.

==========

LIGHT AND SHADE

To add a little more realism to your environments, it is often a good idea to include shaded areas on walls and the "ground" to represent shadows of walls or trees etc. This can easily be done by, for example, putting a slightly darker shade on a triangle, or edited rectangle, up against the lighter wall. Do ensure that you are not adding shadows to animated objects though as the effect can be rather strange once the animation has been effected.

==========

SOUND EFFECTS

The use of multiple sound effects can sometimes give added atmosphere. For example to make the player falls into a pond trigger object: sound of gurgling we could use the and splashing when the following on a selected

```
IF COLLIDED?
THEN SOUND (5)
DELAY (20)
SOUND (5)
```

```
DELAY (30)
SOUND (5)
DELAY (30)
SOUND (9)
ENDIF
```

The sound effects used here are based on the Amiga version.

To make, for example, the sound of a portcullis closing we could have the following condition on our selected trigger object:

```
IF SHOT?
THEN LOOP (40)
DELAY (10)
SOUND (6)
AGAIN
ENDIF
```

Here again we are using the Amiga sound sample bank. This gives a very realistic metallic, clanking sound of a portcullis being lowered.

If we wanted to create the sound effect of a small metalic object being dropped we could enter the following command:

```
IF ACTIVATED?
THEN SOUND (4)
DELAY (50)
SOUND (4)
DELAY (25)
SOUND (4)
DELAY (10)
SOUND (4)
ENDIF
```

By decreasing the DELAY each time we effectively reproduce the sound of the object hitting the ground, bouncing a couple of times and then coming to rest.

==========

VIEWING THE WHOLE AREA

If you wish to view the whole area that you are working on from the top view, instead of clicking on the icon to view one of your created objects from the top, select the "ground" cube (cubeoid 1) and the whole area can be moved over and each object can be viewed within the area. If you are in EDIT mode then each individual object can be selected of editing from above. This is particularly useful if you were creating, for example, a maze such as a hedge maze in a garden. Or if exact positioning of a flat rectangle were required between some other objects.

==========

CHECKING ANIMATIONS

A strategically placed object which has been coloured "invisible" can be very useful for acting as a boundary for an animated object.

****

LISTINGS

Quite a few of you have asked me how the introductory screen was produced in the Kitgame. Once the messages have been printed and the key pressed it is difficult to access the conditions within the startup area.

Here is the listing of the Conditions which may help you to work out what everything means and make life a little easier when creating your own startup sequences.

```
IF VAR>? (V15,0)
THEN SETVAR (0,V40)
SETVAR (0,V248)
SETVAR (2,V35)
GOTO (6,1)
END
ENDIF
REDRAW
WAIT
WAIT
SETVAR (0,V19)
PRINT (" THE 3D KIT GAME",8)
PRINT (" \N\N    CREATED USING",8)
PRINT (" \N\N\N\N THE 3D CONSTRUCTION KIT",8)
PRINT (" \N\N\N\N\N BY INCENTIVE SOFTWARE",8)
PRINT (" \N\N\N\N\N\N\N PRESS A KEY TO START",8)
LOOP 0
WAIT
IF VAR>? (V15,0)
THEN END
ENDIF
IF VAR>? (V19,250)
THEN ELSE AGAIN
ENDIF
PRINT ("PRESS A KEY TO START",7)
REDRAW
WAIT
WAIT
SETVAR (0,19)
PRINT"
```

There now follows a very long message about the Kitgame, how to play it etc which we can see upon loading. We will not reproduce here as it would take up too much space. Once those print commands have finished the program continues immediately with ...

```
LOOP 0
WAIT
IF VAR>? (V15,0)
THEN END
ENDIF
IF VAR>? (V19,1300)
THEN ELSE AGAIN
ENDIF
```

****

THE 3D KIT GAME VARIABLES — 16 BIT

Sincere thanks to the programmers at Incentive for providing this list of all the variables used in the 16 Bit versions of the Kitgame which should help, not only with being able to examine the variables and understand their uses (and to cheat whilst playing the Kitgame) but also in providing ideas on how to program your own variables within your environments.

Variable 31: Time 0-200.

Variable 33: Treasure collected 0-99.

Variable 34: Score 0-99995.

Varibale 35: Text Timer (General Condition).

Variable 38: Cash and Grab counter (Frames).

Variable 39: Cash and Grab counter.

Variable 40: Game State.
0 = Playing,
1 = Lose (Time out),
2 = Lose (Energy out),
3 = Lose (Fell too far),
4 = Lose (Drowned),
5 = Win,
6 = End Game (Endgame sequence and finish).

Variable 41: Area Number (V.08?)

Variable 42: Rope Carried?
0 = Not carried,
1 = Carried.

Variable 43: Time Lapsed in Current Area.

Variable 44: Key Carried?
0 = Not carried,
1 = Carried.

Variable 45: Exit Pass Carried?
0 = Not carried,
1 = Carried.

Variable 46: Scuba Gear Carried?
0 = Not carried,
1 = Worn.

Variable 48: Puzzle Room - Lift X SHIFT.

Variable 49: Puzzle Room - Lift Y SHIFT.

Variable 50: Loop Variable (Multiple Use).

Variable 51: Frames spent in Sea counter.
If 50 then Blackout and go to Desert Island.
If >75 then Drown.

Variable 53: X Coordinate at Beach Head when going to desert island.

Variable 54: Inventory/List Indicator.
0 = Not showing,
1 = First object,
2 = Second object etc.

Variable 55: Object Carried Counter - for inventory list.

Variable 56: Inventory Request.
0 = Not requested,
1 = Requested,
2 = Current showing.

Variable 57: In Sea? (Desert Island).
0 = No,
1 = Yes.

Variable 58: In Sea (Desert Island) Counter.

Variable 60: Animation Start Flag for End Sequence.

Variable 61: Dragon's Nose Shot?
0 = No,
1 = Yes.

Variable 255: Non Resetable Variable - used for Highscore Store.

Variable 254: Z Viewpoint position - used for locking movement controls when animating the view ie. Area 3.

Variable 253: Y Viewpoint position - used for locking movement controls when animating the view ie. Area 3.

Variable 252: X Viewpoint position - used for locking movement controls when animating the view ie. Area 3.

Variable 251: Temporary Variable - Expected to be used for duration of a frame.

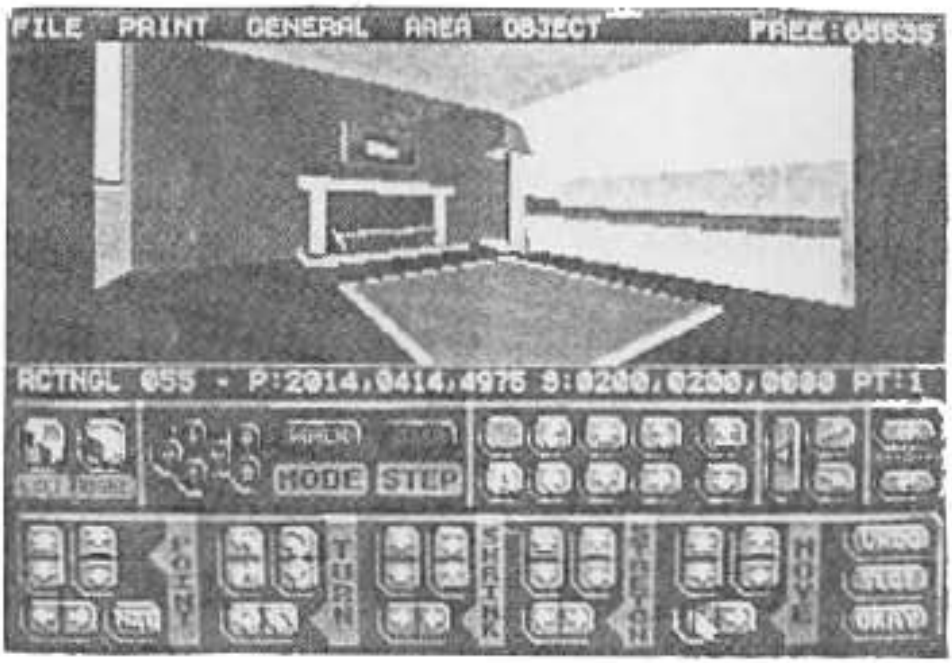
Variable 250: Used to check whether on the boat or not.
0 = Not on the boat,
1 = On the boat.

Variable 249: Used to check whether in the sea or not.
0 = Not in the sea.
1 = In the sea.

Variable 248: Cheat on?
0 = No,
1 = Yes.

Variable 247: Speed Trap On? (PC).
0 = No,
1 = Yes.

A list of the variables used within the 8 Bit version of the 3D Kitgame will be published in the newsletter when those versions of the program become available.



THE 3D CONSTRUCTION KIT USERS CLUB
[illegible]